WAVOO WAJEEHA WOMEN'S COLLEGE
OF ARTS & SCIENCE - KAYALPATNAM
(Affiliated to Manonmanium Sundaranar University, Tirunelveli)
Run by : Wavoo SAR Educational Trust
(minority institution)

Department of Information Technology
Statistics with Python

Module 4: **Math Module in Python** -Euler's number, Pi, Trigonometric and Logarithmic Functions, Numeric Functions – Ceil, Floor, Factorial, GCD,LCM

Python - Math Module

Some of the most popular mathematical functions are defined in the math module. These include trigonometric functions, representation functions, logarithmic functions, angle conversion functions, etc. In addition, two mathematical constants are also defined in this module.

The math module has a set of methods and constants.

Math Methods

Method	Description
math.acos()	Returns the arc cosine of a number
math.acosh()	Returns the inverse hyperbolic cosine of a number
math.asin()	Returns the arc sine of a number
math.asinh()	Returns the inverse hyperbolic sine of a number
math.atan()	Returns the arc tangent of a number in radians
math.atan2()	Returns the arc tangent of y/x in radians
math.atanh()	Returns the inverse hyperbolic tangent of a number
math.ceil()	Rounds a number up to the nearest integer
math.comb()	Returns the number of ways to choose k items from n items without repetition and order
math.copysign()	Returns a float consisting of the value of the first parameter and the sign of

	the second parameter
math.cos()	Returns the cosine of a number
math.cosh()	Returns the hyperbolic cosine of a number
math.degrees()	Converts an angle from radians to degrees
math.dist()	Returns the Euclidean distance between two points (p and q), where p and q are the coordinates of that point
math.erf()	Returns the error function of a number
math.erfc()	Returns the complementary error function of a number
math.exp()	Returns E raised to the power of x
math.expm1()	Returns E^x - 1
math.fabs()	Returns the absolute value of a number
math.factorial()	Returns the factorial of a number
math.floor()	Rounds a number down to the nearest integer
math.fmod()	Returns the remainder of x/y
math.frexp()	Returns the mantissa and the exponent, of a specified number
math.fsum()	Returns the sum of all items in any iterable (tuples, arrays, lists, etc.)
math.gamma()	Returns the gamma function at x
math.gcd()	Returns the greatest common divisor of two integers
math.hypot()	Returns the Euclidean norm
math.isclose()	Checks whether two values are close to each other, or not
math.isfinite()	Checks whether a number is finite or not
math.isinf()	Checks whether a number is infinite or not
math.isnan()	Checks whether a value is NaN (not a number) or not
math.isqrt()	Rounds a square root number downwards to the nearest integer
math.ldexp()	Returns the inverse of math.frexp() which is x * (2**i) of the given numbers x and i
math.lgamma()	Returns the log gamma value of x
math.log()	Returns the natural logarithm of a number, or the logarithm of number to

	base
math.log10()	Returns the base-10 logarithm of x
math.log1p()	Returns the natural logarithm of 1+x
math.log2()	Returns the base-2 logarithm of x
math.perm()	Returns the number of ways to choose k items from n items with order and without repetition
math.pow()	Returns the value of x to the power of y
math.prod()	Returns the product of all the elements in an iterable
math.radians()	Converts a degree value into radians
math.remainder()	Returns the closest value that can make numerator completely divisible by the denominator
math.sin()	Returns the sine of a number
math.sinh()	Returns the hyperbolic sine of a number
math.sqrt()	Returns the square root of a number
math.tan()	Returns the tangent of a number
math.tanh()	Returns the hyperbolic tangent of a number
math.trunc()	Returns the truncated integer parts of a number

Math Constants

Constant	Description
math.e	Returns Euler's number (2.7182...)
math.inf	Returns a floating-point positive infinity
math.nan	Returns a floating-point NaN (Not a Number) value
math.pi	Returns PI (3.1415...)
math.tau	Returns tau (6.2831...)

PI:

Pi is a well-known mathematical constant, which is defined as the ratio of the circumference to the diameter of a circle and its value is 3.141592653589793.

Example:

```
>>> import math
>>>math.pi
3.141592653589793
```

Euler's number

Well-known mathematical constant defined in the math module is **e**. It is called **Euler's number** and it is a base of the natural logarithm. Its value is 2.718281828459045.

Example:

```
>>> import math
>>> math.e
2.718281828459045
```

Trigonometric Functions-

The math module contains functions for calculating various trigonometric ratios for a given angle. The functions (sin, cos, tan, etc.) need the angle in radians as an argument. We, on the other hand, are used to express the angle in degrees.

The math module presents two angle conversion functions: degrees() and radians(), to convert the angle from degrees to radians and vice versa. For example, the following statements convert the angle of 30 degrees to radians and back (Note: π radians is equivalent to 180 degrees).

Example: Math Radians and Degrees

```
>>> import math
>>> math.radians(30)
0.5235987755982988
>>> math.degrees(math.pi/6)
29.999999999999996
```

The following statements show sin, cos and tan ratios for the angle of 30 degrees (0.5235987755982988 radians)

Example: sin, cos, tan Calculation

```
>>> import math
>>> math.sin(0.5235987755982988)
0.49999999999999994

>>> math.cos(0.5235987755982988)
0.8660254037844387

>>> math.tan(0.5235987755982988)
0.5773502691896257
```

Logarithmic Functions

math.log()

The math.log() method returns the natural logarithm of a given number. The natural logarithm is calculated to the base e.

Example:

```
>>> import math
>>>math.log(10)
2.302585092994046
```

math.log10()

The math.log10() method returns the base-10 logarithm of the given number. It is called the standard logarithm.

Example:

```
>>> import math
>>>math.log10(10)
1.0
```

Numeric Functions - Ceil, Floor, Factorial, GCD,LCM

The **ceil()** function approximates the given number to the smallest integer, greater than or equal to the given floating point number. The **floor()** function returns the largest integer less than or equal to the given number.

Example:

```
>>> import math
>>> math.ceil(4.5867)
5
>>> math.floor(4.5687)
4
```

math.sqrt()-The math.sqrt() method returns the square root of a given number.

Example: Square Root

```
>>> import math
>>> math.sqrt(100)
10.0
>>> math.sqrt(3)
1.7320508075688772
```

math.exp()-The math.exp() method returns a float number after raising e to the power of the given number. In other words, exp(x) gives e**x.

Example: Exponent

```
>>> import math
>>>math.exp(10)
```

```
22026.465794806718
```

This can be verified by the exponent operator.

Example: Exponent Operator **

```
>>> import math
>>>math.e**10
22026.465794806703
```

math.pow()-The math.pow() method receives two float arguments, raises the first to the second and returns the result. In other words, pow(4,4) is equivalent to 4**4.

Example: Power

```
>>> import math
>>> math.pow(2,4)
16.0
>>> 2**4
16
```

math.factorial()

The math.factorial() method returns the factorial of a number.
Note: This method only accepts positive integers.

The factorial of a number is the sum of the multiplication, of all the whole numbers, from our specified number down to 1. For example, the factorial of 6 would be 6 x 5 x 4 x 3 x 2 x 1 = 720

```
#Import math Library
import math
#Return factorial of a number
print(math.factorial(9))
print(math.factorial(6))
print(math.factorial(12))
```

math.gcd()

The math.gcd() method returns the greatest common divisor of the two integers *int1* and *int2*. GCD is the largest common divisor that divides the numbers without a remainder.GCD is also known as the highest common factor (HCF).

Tip: gcd(0,0) returns 0.

Syntax
math.gcd(*int1*, *int2*)

Example

Find the greatest common divisor of the two integers 3 and 6.

```
#Import math Library
import math
#find the  the greatest common divisor of the two integers
print (math.gcd(3, 6))
print (math.gcd(6, 12))
print (math.gcd(12, 36))
print (math.gcd(-12, -36))
print (math.gcd(5, 12))
print (math.gcd(10, 0))
print (math.gcd(0, 34))
print (math.gcd(0, 0))
```

OUTPUT:

```
3
6
12
12
1
10
34
0
```

math.lcm()- math.lcm(*integers) function returns the Least Common Multiple (LCM) of the integers passed as arguments**.**

Syntax

The syntax to call lcm() function is

```
math.lcm(*integers)
```

If no integer value is passed as argument to lcm(), then the return value is 1.
If only one integer is passed to lcm(), then then same value is returned as LCM.
If any non-integral value is passed as argument to lcm(), then the function raises TypeError.

Examples

LCM of two integers: 10 and 25.

```
import math
result = math.lcm(10, 25)
print('lcm() :', result)
```

Output

```
lcm() : 50
```

Sources:

https://www.w3schools.com/python/module_math.asp